package com.twitter.search.earlybird.util;

import java.util.concurrent.Callable;

import com.google.common.base.Stopwatch;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public final class ActionLogger {

private static final Logger LOG = LoggerFactory.getLogger(ActionLogger.class);

private ActionLogger() {

}

/\*\*

\* Run a function, logging a message at the start and end, and the time it took.

\*/

public static <T> T call(String message, Callable<T> fn) throws Exception {

LOG.info("Action starting: '{}'.", message);

Stopwatch stopwatch = Stopwatch.createStarted();

try {

return fn.call();

} catch (Throwable e) {

LOG.error("Action failed: '{}'.", message, e);

throw e;

} finally {

LOG.info("Action finished in {} '{}'.", stopwatch, message);

}

}

/\*\*

\* Run a function, logging a message at the start and end, and the time it took.

\*/

public static void run(String message, CheckedRunnable fn) throws Exception {

call(message, () -> {

fn.run();

return null;

});

}

@FunctionalInterface

public interface CheckedRunnable {

/\*\*

\* A nullary function that throws checked exceptions.

\*/

void run() throws Exception;

}

}